

Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

- 1 1. (Presently Amended) A method for managing data volumes from a management
2 terminal in a distributed computer system having a host computer system with at
3 least one storage device connected to the computer system by driver software,
4 the method comprising:
 - 5 (a) inserting an interface layer between the driver software and the storage
6 device, the interface layer exporting a platform dependent API comprising
7 a plurality of API methods that can be called to control and controlling data
8 passing between the driver software and the storage device;
 - 9 (b) running, in the host computer system, management facade software that
10 converts the interface layer API to receives calls to platform-independent
11 method calls methods and generates at least one API method call to the
12 interface layer in order to execute the platform-independent method calls;
 - 13 (c) running, in the host, a federated bean that discovers data volumes in the
14 storage device and generates platform-independent method calls to the
15 management facade to control the interface layer via the plurality of API
16 methods; and
 - 17 (d) controlling the federated bean to display and configure the data volumes.
- 1 2. (Original) The method of claim 1 wherein step (d) comprises controlling the
2 federated bean with a command line interface.
- 1 3. (Original) The method of claim 1 wherein step (d) comprises controlling the
2 federated bean with a graphical user interface.

1 4. (Original) The method of claim 1 wherein step (a) comprises inserting a SCSI
2 terminal emulation interface layer between the driver software and the storage
3 device, which interface layer makes the storage device appear as a SCSI device.

1 5. (Original) The method of claim 1 wherein step (a) comprises inserting a storage
2 volume interface layer between the driver software and the storage device,
3 inserting an additional data service interface layer between the storage volume
4 interface layer and the storage device and using the storage volume interface
5 layer to divert information passing between the driver software and the storage
6 device to the additional data service layer.

1 6. (Original) The method of claim 1 wherein step (c) comprises:
2 (c1) displaying all data volumes on the host with information regarding the data
3 volumes displayed in a uniform manner.

1 7. (Original) The method of claim 6 wherein the data volumes on the host have
2 various volume types and step (c1) comprises displaying all data volumes of
3 each volume type together wherein information regarding each data volume type
4 is presented in a uniform manner.

1 8. (Original) The method of claim 1 further comprising:
2 (e) running a distributed management framework in the computer system, the
3 distributed management framework having a lookup service; and
4 (f) using the lookup service to locate the federated bean.

1 9. (Original) The method of claim 1 wherein the federated bean maintains a list of
2 users for each of the data volumes and the method further comprises:
3 (g) registering each user of one of the data volumes with the federated bean
4 when the each user requests use of the one data volume; and

(h) upon registration, entering the data volume user into a list maintained for the one data volume.

10. (Original) The method of claim 1 further comprising:

- (i) retrieving a list of users for a specific volume from the federated bean; and
- (j) contacting each user on the list to determine how the each user is using the specific volume.

11. (Original) The method of claim 10 wherein each user has a method describing how the each user is using the specific volume and wherein step (j) comprises contacting each user on the list and running the method in the each user.

12. (Presently Amended) Apparatus for managing data volumes from a management terminal in a distributed computer system having a host computer system with at least one storage device connected to the computer system by driver software, the apparatus comprising:

an interface layer that is inserted between the driver software and the storage device, the interface layer exporting a platform dependent API comprising a plurality of API methods that can be called to control and controlling data passing between the driver software and the storage device;

management facade software in the host computer system that ~~converts the interface layer API to~~ receives calls to platform-independent method calls methods and generates at least one API method call to the interface layer in order to execute the platform-independent method calls;

a federated bean in the host that discovers data volumes in the storage device and generates platform-independent method calls to the management facade to control the interface layer via the plurality of API methods; and

a presentation program that controls the federated bean to display and configure the data volumes.

- 1 13. (Original) The apparatus of claim 12 wherein the presentation program
2 comprises a command line interface.
- 1 14. (Original) The apparatus of claim 12 wherein the presentation program
2 comprises a graphical user interface.
- 1 15. (Original) The apparatus of claim 12 wherein the interface layer comprises a
2 SCSI terminal emulation interface layer that makes the storage device appear as
3 a SCSI device.
- 1 16. (Original) The apparatus of claim 12 wherein the interface layer comprises:
2 a storage volume interface layer located between the driver software and
3 the storage device; and
4 an additional data service interface layer located between the storage
5 volume interface layer and the storage device and wherein the storage volume
6 interface layer is used to divert information passing between the driver software
7 and the storage device to the additional data service layer.
- 1 17. (Original) The apparatus of claim 12 wherein the federated bean comprises a
2 logical disk aggregator that retrieves information from all data volumes on the
3 host.
- 1 18. (Original) The apparatus of claim 17 wherein the data volumes on the host have
2 various volume types and the logical disk aggregator comprises mechanisms that
3 retrieve information from each volume type.
- 1 19. (Original) The apparatus of claim 12 wherein the computer system has a
2 distributed management framework with a lookup service running therein; and
3 the apparatus further comprises a mechanism in the lookup service that locates
4 the federated bean.

1 20. (Original) The apparatus of claim 12 wherein the federated bean comprises a list
2 of users for each of the data volumes and a mechanism for registering each user
3 of one of the data volumes with the federated bean when the each user requests
4 use of the one data volume by entering the data volume user into a list
5 maintained for the one data volume.

1 21. (Original) The apparatus of claim 12 wherein the federated bean further
2 comprises a mechanism that retrieves a list of users for a specific volume from
3 the federated bean and a mechanism that contacts each user on the list to
4 determine how the each user is using the specific volume.

1 22. (Original) The apparatus of claim 21 wherein each user has a method describing
2 how the each user is using the specific volume and wherein the mechanism that
3 contacts each user comprises a mechanism for calling the method in the each
4 user.

1 23. (Presently Amended) A computer program product for managing data volumes
2 from a management terminal in a distributed computer system having a host
3 computer system with at least one storage device connected to the computer
4 system by driver software, the computer program product comprising a computer
5 usable medium having computer readable program code thereon, including:

6 interface layer program code for insertion between the driver software and
7 the storage device, the interface layer program code exporting a platform
8 dependent API comprising a plurality of API methods that can be called to control
9 and controlling data passing between the driver software and the storage device;

10 management facade software that ~~converts the interface layer API to~~
11 receives calls to platform-independent method calls methods and generates at
12 least one API method call to the interface layer in order to execute the platform-
13 independent method calls;

14 federated bean program code that discovers data volumes in the storage
15 device and generates platform-independent method calls to the management
16 facade to control the interface layer via the plurality of API methods; and
17 a presentation program that controls the federated bean to display and
18 configure the data volumes.

1 24. (Original) The computer program product of claim 23 wherein the presentation
2 program comprises a command line interface.

1 25. (Original) The computer program product of claim 23 wherein the presentation
2 program comprises a graphical user interface.

1 26. (Presently Amended) A computer data signal embodied in a carrier wave for
2 managing data volumes from a management terminal in a distributed computer
3 system having a host computer system with at least one storage device
4 connected to the computer system by driver software, the computer data signal
5 comprising:

6 interface layer program code for insertion between the driver software and
7 the storage device, the interface layer program code exporting a platform
8 dependent API comprising a plurality of API methods that can be called to control
9 ~~and controlling~~ data passing between the driver software and the storage device;

10 management facade software that ~~converts the interface layer API to~~
11 receives calls to platform-independent method calls methods and generates at
12 least one API method call to the interface layer in order to execute the platform-
13 independent method calls;

14 federated bean program code that discovers data volumes in the storage
15 device and generates platform-independent method calls to the management
16 facade to control the interface layer via the plurality of API methods; and
17 a presentation program that controls the federated bean to display and
18 configure the data volumes.